



#ШПАРГАЛОЧКИ

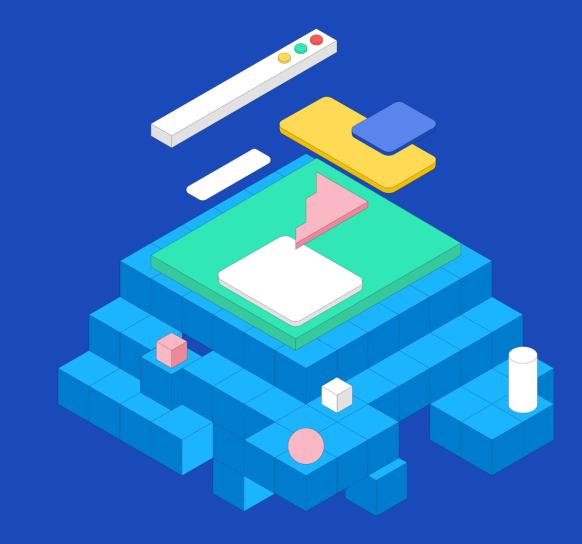
СОЗДАНИЕ САЙТОВ FRONT-END РАЗРАБОТКА

Дополнительный уровень

Материалы подготовлены отделом методической разработки

Больше полезных материалов и общения в нашем комьюнити в Telegram: https://t.me/hw_school











Перенос данных, знакомство с BLL







BLL

BLL (**Buisiness Logic Layout**) или слой бизнес-логики - это данные, которые использует реакт-приложение.

Например, компонента - это не данные, компонента формирует **UI** (**User Interface**) - пользовательский интерфейс (текстовые поля, кнопки и так далее). А вот то, что компонента принимает в **props** (имя пользователя, его id) - это уже данные.







BLL

Чтобы разделить бизнес-логику и пользовательский интерфейс, данные нужно хранить в массиве объектов:

А в компоненте - только использовать эти данные:

<DialogsItem name={dialogNames[0].name} id={dialogNames[0].id} />







map()

тар - метод массивов, который позволяет "перебрать" массив и поочереди вытащить из него каждый элемент. С помощью этого метода можно вывести сколько угодно компонент (например, собеседников) всего в одну строку кода:

{dialogNames.map((e) => <DialogsItem name={e.name} id={e.id} />)}

Тут тар принимает **стрелочную функцию** с параметром **e** - в него по очереди запишется каждый элемент массива **dialogItems**. Эта функция возвращает **новый массив** с компонентами **DialigsItem** и соответствующими атрибутами.







Документация проекта

Чтобы не запутаться в сложном проекте, полезно иметь схему, описывающую его структуру.

Сейчас точка входа в нашем проекте - это **index.js**, отрисовывающая компоненту **App**.

App в свою очередь принимает Header, Navbar и Profile или Dialogs.

Profile отрисовывает компоненты ProfileInfo и Posts (которая заполняется из Post), Dialogs - DialogItem и Messages.







Документация проекта

